



Prediction of Appointment No Show

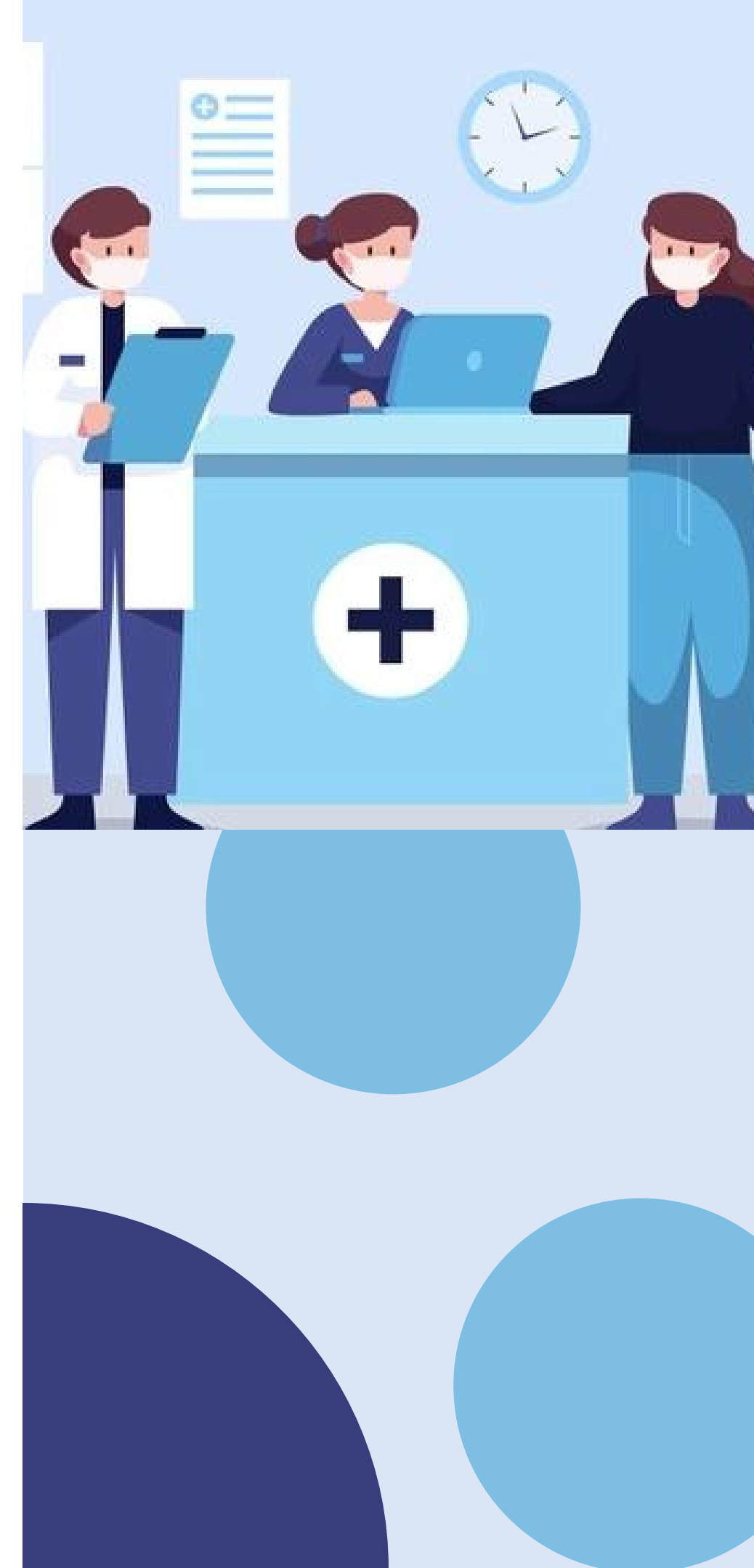
Presented by [Abeer Almdani](#) | [Rana Alzahrani](#)

Outlines

- 01** Introduction and problem statement
- 02** Design
- 03** Data
- 04** Feature Engineering
- 05** Algorithms and Models
- 06** Tools

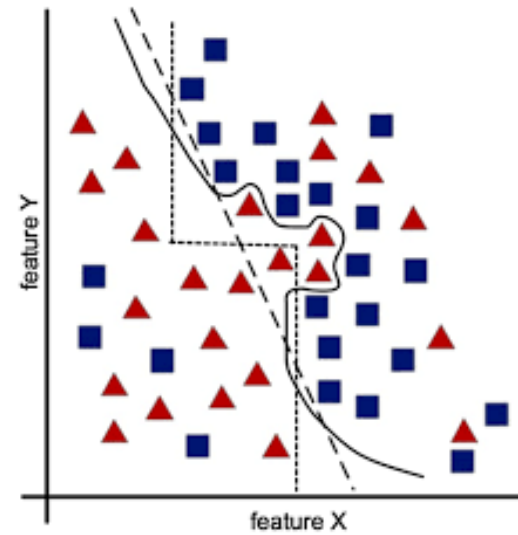
Introduction and problem statement

Patients make appointments at clinics or hospitals to be checked by a doctor. Some of the patients do not show up for their appointments. This results in loss of valuable resources in terms of physician time and staffing allocation which could have been used more productively. We will predict if the person shows up or not using a few classification algorithms.



Design

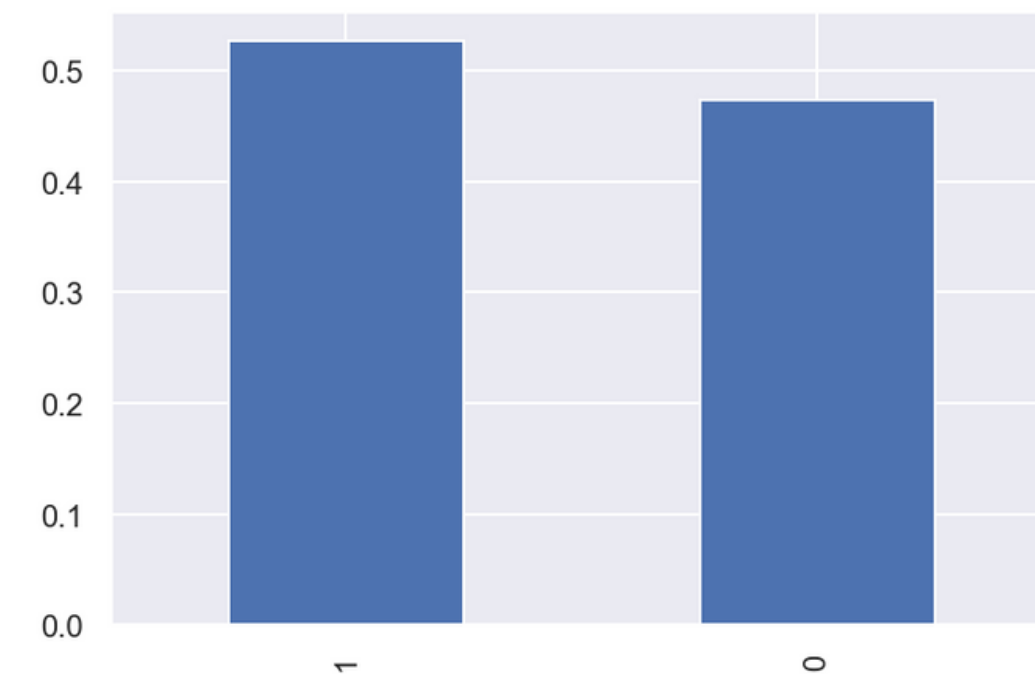
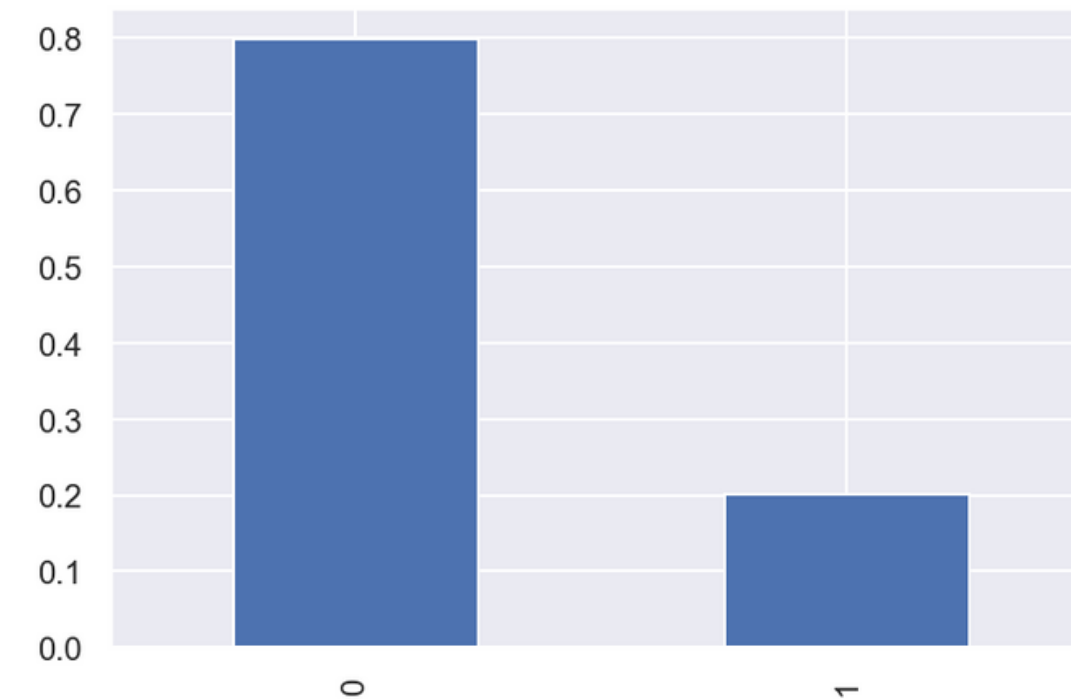
- Data source
- Classification Moduls
- Evaluation



Data

EDA and Feature Engineering

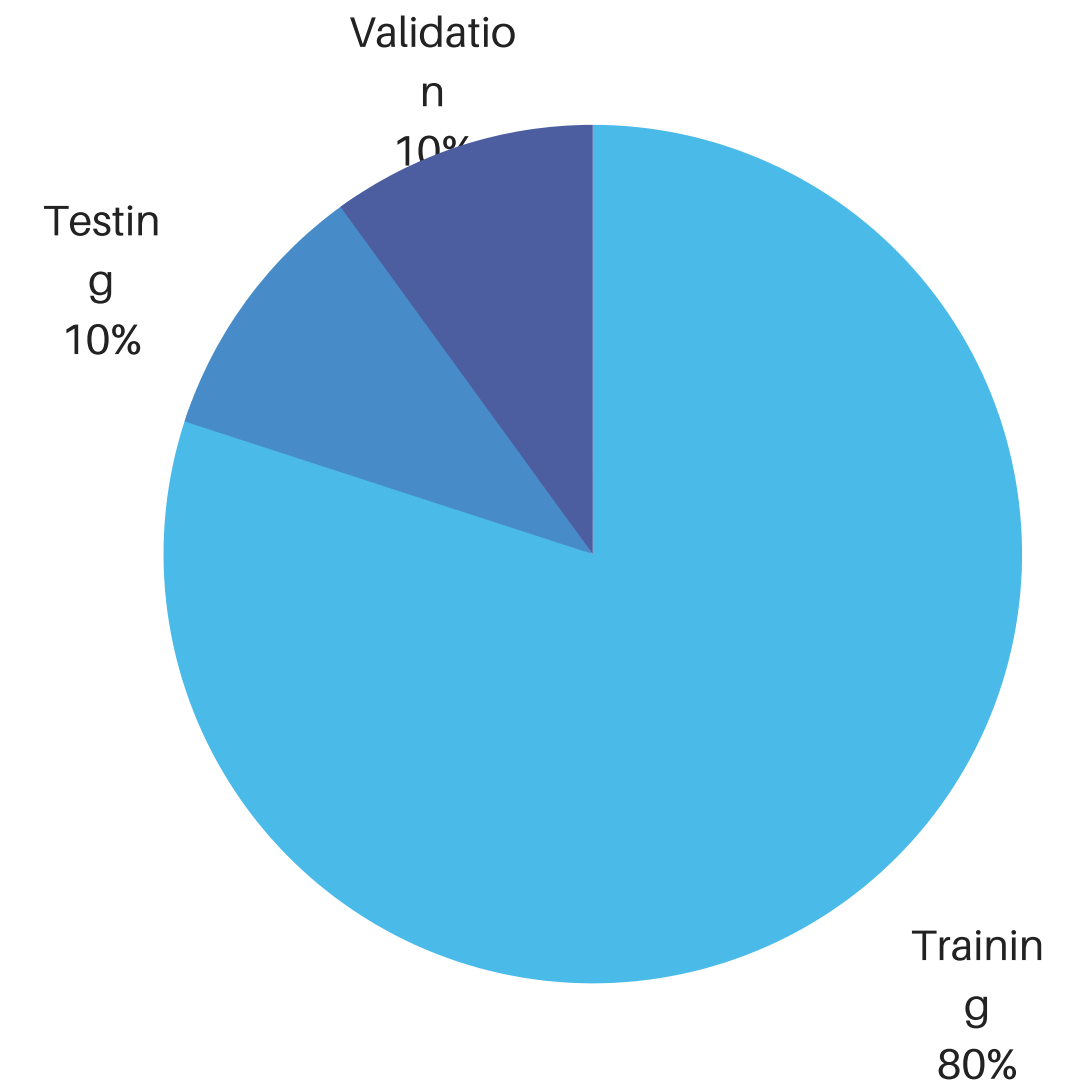
- String Binary (yes,no) --> (0,1)
- String type--> int
- Age > 0
- Neighbourhood --> dummies (81 cols)
- **Adding features**
 - Days
 - Morning
 - Appointment Day
 - Appointment Month
 - Scheduled Day
 - Scheduled Month
 - Week Day
- **Deleting useful features**
 - Scheduled Date
 - Appointment Date
 - Appointment ID
 - Patient ID



Data

- **Data Size before Engineering and UnderSampling:**
 - Number of rows: 110,528
 - Number of columns: 14
- **Data Size after Engineering and UnderSampling:**
 - Number of rows: 74,175
 - Number of columns: 98
- Target = No Show
- Features = Gender, ScheduledDay, AppointmentDay Age, Neighbourhood, Scholarship, Hipertension, Diabetes, Alcoholism, Handcap, SMS_received, Scheduled_Day, Scheduled_Month, Weekday, days

Splinted Data



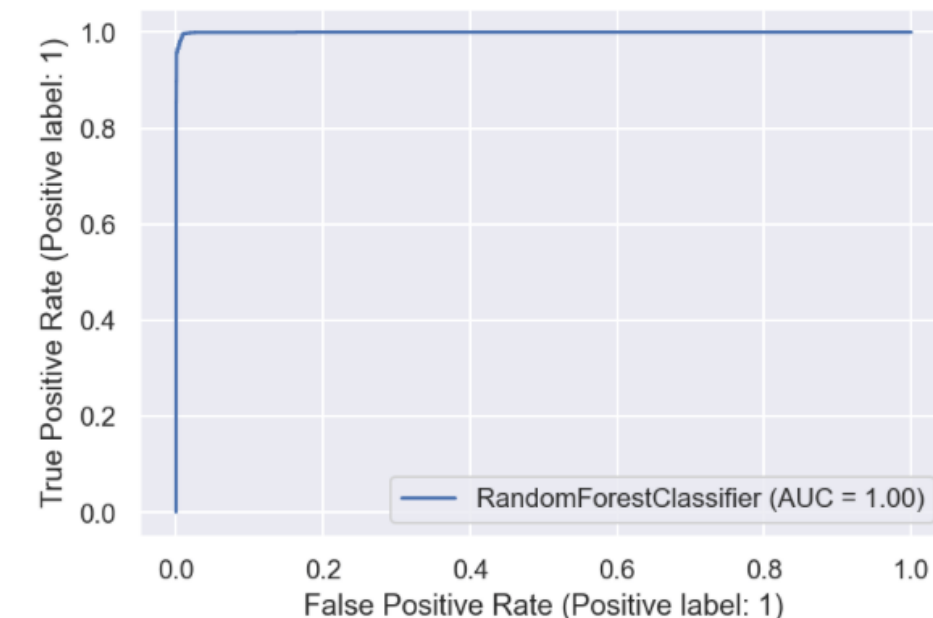
Algorithms and Models

Modules	AccuracyTr	Validation Score	Grid search
BaseLine	79.80	79.869	C=1
Logistic regression	79.52%	79.65%	C=1
Knn	99.36%	80.25%	n_estimators= 161
Decision Tree	86.78%	77.75%	max_depth=21
Random Forest	99.29%	79.36%	n_neighbors=55
Extra Trees	99.29%	78.04%	-
XGBClassifier	87.33%	80.25%	-
Polynomial 2	99.20%	99.09%	-

**55-nearest neighbors
algorithm
is the best algorithm**

Standard classification metric

- Score of train set: 0.99358
- Score of validation set: 0.8025
- Score of test set: 0.79861
- Accuracy: 0.79861
- Precision: 0.33333
- Recall: 0.00045
- F1_Score: 0.0009



dealing with over and under fitting

- Standard Scaler
- Adding Features
- Delateing Features
- minimize and maximize training and validation set
- Validation score increase 5.85



Tools

- Sklearn
- Numpy
- Pandas
- Seaborn
- Python
- Anaconda
- Canva
- Github
- Zoom
- Jupyter notebook





Thanks

Any question?